Computer Communications 32 (2009) 1600–1612

Contents lists available at ScienceDirect

Computer Communications

journal homepage: www.elsevier.com/locate/comcom

Using cross-layer metrics to improve the performance of end-to-end handover mechanisms

John Fitzpatrick [a,*], Séan Murphy [a], Mohammed Atiquzzaman [b], John Murphy [a]

[a] School of Computer Science and Informatics, University College Dublin, Belfield, Dublin 4, Ireland
[b] School of Computer Science, University of Oklahoma, Norman, OK 73019-6151, USA

ARTICLE INFO

Article history:
Received 23 April 2009
Received in revised form 3 June 2009
Accepted 6 June 2009
Available online 13 June 2009

Keywords:
Echo
SCTP
VoIP
E-Model
Sigma

ABSTRACT

Network centric handover solutions for all IP wireless networks usually require modifications to network infrastructure which can stifle any potential rollout. This has led researchers to begin looking at alternative approaches. Endpoint centric handover solutions do not require network infrastructure modification, thereby alleviating a large barrier to deployment. Current endpoint centric solutions capable of meeting the delay requirements of Voice over Internet Protocol (VoIP) fail to consider the Quality of Service (QoS) that will be achieved after handoff. The main contribution of this paper is to demonstrate that QoS aware handover mechanisms which do not require network support are possible. This work proposes a Stream Control Transmission Protocol (SCTP) based handover solution for VoIP called Endpoint Centric Handover (ECHO). ECHO incorporates cross-layer metrics and the ITU-T E-Model for voice quality assessment to accurately estimate the QoS of candidate handover networks, thus facilitating a more intelligent handoff decision. An experimental testbed was developed to analyse the performance of the ECHO scheme. Results are presented showing both the accuracy of ECHO at estimating the QoS and that the addition of the QoS capabilities significantly improves the handover decisions that are made.

1. Introduction

There currently exists a multitude of disparate wireless access networks with very little integration between them. As the number of wireless access networks grows there will be an increasing demand for users to have ubiquitous access to services across these multiple networks. For this integration to take place, handover solutions capable of meeting the handover requirements of delay sensitive applications such as VoIP must be developed.

Much work has been done in the area of IP mobility with most approaches focusing on network oriented solutions such as Mobile IP (MIP). However, despite the widespread availability of MIP capable network components for some time the technology has seen very little deployment. One of the main deployment problems is that network oriented solutions require a large number of networks to be upgraded before the mobility solution can become useful.

Further, network oriented solutions tend to have an operator centric approach to mobility. Although they can provide handovers between heterogeneous Radio Access Networks (RANs), each access network must be part of the same network operators infrastructure. This has led researchers to begin looking at alternative approaches to host mobility.

By shifting the complexity out of the network and into the mobile devices, endpoint centric handover solutions can be developed. Endpoint based solutions require no network infrastructure modifications and therefore have fewer barriers to deployment. *S*eamless *I*P diversity based *G*eneralized *M*obility *A*rchitecture (SIGMA) is one such solution, which is based on the SCTP transport protocol [1]. SIGMA leverages the multihoming capability of SCTP to effect seamless handovers capable of meeting the strict delay requirement of real time applications such as VoIP [2]. SIGMA handover decisions are based on a simple comparison of the Received Signal Strength (RSS) from each available access network. The QoS in terms of delay, loss and jitter that each network can offer is not considered. This can result in handovers to networks that cannot support the QoS required by the application if, for example, a network was congested.

In this paper, ECHO, is proposed. ECHO enhances the SIGMA concept to mobility by considering the QoS that will be obtained from each of the candidate handover networks. ECHO uses a cross-layer approach to obtain metrics that directly effect the QoS of a VoIP call. The ITU-T E-Model is used in real time to map these metrics to a user perceived quality metric for VoIP called the Mean Opinion Score (MOS). The independent MOS scores for each access network are then used to make handover decisions which result in improved VoIP performance.

The rest of this paper is structured as follows. Section 2 gives an overview of related work. Section 3 describes aspects of SCTP and

* Corresponding author. Tel.: +353 1 7162425.
E-mail address: john.fitzpatrick@ucd.ie (J. Fitzpatrick).

0140-3664/$ - see front matter © 2009 Elsevier B.V. All rights reserved.
doi:10.1016/j.comcom.2009.06.005

the E-Model pertinent to the proposed work. Section 4 describes the cross-layer metrics used by the ECHO handover mechanism. The design and implementation of ECHO is then discussed in Section 5. Section 6 describes the experimental testbed used to analyse the performance of ECHO followed a discussion of the results obtained in Section 7. The paper is concluded in Section 8.

2. Literature review

There has been much work done in the area of IP mobility with many different approaches proposed. The most common mobility scheme is MIP [3,4]. MIP allows each Mobile Node (MN) to use two IP addresses, a static address known as the home address present at the home network and a Care of Address (CoA) that changes with each new point of attachment to each foreign network. By maintaining bindings between the two addresses MIP allows the MN to be always reachable through its home address. This is accomplished through the addition of new network nodes. Specifically, MIP requires the addition of a Home Agent (HA) in the network where a MN is normally resident and a Foreign Agent (FA) in any network a MN may visit. The purpose of the FA is to assign the IP address that will serve as the MNs CoA and to inform the HA of the MNs new Internet Protocol (IP) address. A tunnel is established between the FA and the MNs home network so that the MN is reachable via its HA; unfortunately this leads to increased link delay and problems with scalability. The major drawback in using MIP is the required introduction of new network nodes meaning that large scale deployment is required before the technology becomes useful.

There has however been a large amount of work done in improving the performance of MIP. MIPv6 [5] improves on MIP by requiring less infrastructure modifications and solving triangular routing problems. Specifically, in MIPv6 no special local routers are needed as foreign agents. This reduces a barrier to deployment as less network infrastructure modifications are needed. Also, the problem of triangular routing has been addressed with the use of binding updates. Although MIPv6 overcomes some of the problems of traditional MIP it assumes the widespread use of IPv6 as well as requiring the introduction of new network nodes to act as HAs. Also, it has been shown that the time taken to perform a handover can be too great to meet the strict time constraints of real time applications such as VoIP [6].

In traditional MIP and MIPv6 a MN must break the connection with the existing AP before obtaining an address with a new Access Point (AP) which leads to high handover latency. To address this handover latency Fast handovers for MIPv6 (FHMIP) [7] has been proposed. The main aim of this protocol is to allow a MN to obtain a CoA with a new AP before disconnecting from its current AP. This allows the new CoA to be used immediately on connection with the new AP thereby decreasing the handover latency. Unfortunately this approach still requires modifications to network infrastructure. Additionally, since a CoA must be allocated to a MN before it connects, additional signalling is required by the neighbour discovery mechanism which can result in scalability issues.

An alternate approach known as Hierarchical Mobile IP (HMIP) [8] improves both the handoff performance and scalability of MIP by using a two level hierarchy separating local mobility from global mobility. Mobility Servers (MSs) are introduced to handle local mobility while MIPv6 is used for global mobility. Although this solution greatly improves the handoff performance of MIP, it requires even greater infrastructure modifications with the addition of a mobility server at each level of the hierarchy. Also, since MIPv6 is used for global mobility it cannot meet the delay requirements of real time applications [6].

Other approaches have also been proposed to alleviate some of the problems associated with MIP. For example, Cellular IP (CIP) [9] is an IP mobility solution that incorporates some cellular system features. Local mobility, is handled by cellular IP protocols while MIP is used to support global mobility. Another approach called Handoff-Aware Wireless Access Internet Infrastructure (HAWAII) [10] divides the network into a hierarchy of domains in which a MNs IP address does not change as it moves through base stations within the same domain. However, since both of these hierarchical approaches require a large amount of infrastructure modification they suffer from the same deployment problems as MIP. Also, as MIPv6 is used as the handover protocol between each hierarchical level the handover latency is still an issue.

Another MIP based solution is Proxy MIPv6 (PMIPv6) [11]. PMIPv6 is designed to provide MIP functionality, within a defined network, to nodes without MIP client functionality (No client side MIP support is assumed). Although PMIPv6 does partially solve some of the problems of deployment associated with MIP it still requires new network infrastructure and may suffer from the same limited deployment issues that have affected MIP.

More recently there has been increasing interest in approaches which require no network support – end-to-end approaches. End-to-end solutions move intelligence from the network to the mobile terminals having the advantages of requiring fewer or no network modifications and easing rollout [12].

Transmission Control Protocol (TCP) Migrate [13] is an approach to end-to-end internet host mobility. It focuses on the issue of continuing an existing TCP session without having to re-establish the TCP connection. A new option is proposed in SYN packets that identifies the packet as part of a previous TCP connection allowing a MN to restart an open TCP connection from a new point of attachment. The advantage of this scheme is that it does not require any network infrastructure modifications, only the TCP stack of each node need be upgraded. However, since this mobility solution only works with TCP and can suffer from significant handover delays it is unsuitable for real time applications such as VoIP.

Another transport layer based approach is Mobile SCTP (MSCTP) [14]. MSCTP leverages the ability of SCTP to have multiple IP addresses per association.[1] MSCTP utilises a feature of SCTP called the Dynamic Address Reconfiguration (DAR) extension which allows a MN to dynamically switch between available access networks thereby effecting seamless handovers. MSCTP suggests the use of Session Initiation Protocol (SIP) or MIP to deal with location management but focuses on using MIP. Although only the MN needs to be MIP enabled, MSCTP still requires network modifications to implement a HA. Consequently, it suffers from the same network modification requirements problems as MIP. Also, a handover decision is made simply based on the RSS at the MN and does not incorporate any QoS metrics.

3. Background – SCTP and the E-Model

In order to fully understand the proposed mechanism some knowledge of SCTP and its extensions are required. Further, an understanding of the E-Model used for calculating VoIP QoS is also required. This section gives an overview of the pertinent features of each.

3.1. SCTP

SCTP is a message based[2] end-to-end connection oriented transport layer protocol approved by the Internet Engineering Task Force

[1] A central concept in SCTP is the definition of an association. An *association* in SCTP is analogous to a connection in TCP.

[2] SCTP is message oriented and supports framing of individual message boundaries as opposed to TCPs byte oriented approach which does not preserve any implicit structure within a transmitted byte stream.

(IETF) [15]. SCTP operates in a similar manner to TCP offering reliable, sequenced transport of messages, but also offers extra capabilities. Indeed SCTP inherited many of the core features of TCP such as congestion control and retransmission. SCTP now has a more broad scope as a general transport layer protocol. Consequently applications can now benefit from SCTP features allowing higher performance and reliability than other protocols.

The most important features of SCTP relevant to the work proposed in this paper are multihoming and multistreaming. Multihoming allows a single association to span multiple IP addresses. Each IP address can be bound to a separate IP interface connected to different physical networks. During the initial setup of an association between two endpoints, one of the IP addresses at each endpoint is designated the primary address. All communication to that endpoint is routed to the primary until a network failure is detected or an upper layer specifically requests the use of an alternate IP address.

The multistreaming feature of SCTP allows independent streams of data to be transmitted across a single association with no reliance on the delivery order of packets in other streams. Multistreaming is used in SCTP to solve the TCP problem of Head of Line (HOL) blocking that arises from TCPs strict byte ordered delivery. Although SCTP has many other features discussion of these is not relevant to the proposed scheme and is beyond the scope of this paper.

3.1.1. SCTP – dynamic address reconfiguration

In the original variant of SCTP the primary IP address can only be changed to an alternate address that was included in the set of available addresses exchanged at setup. Therefore, a node must in advance have an IP address with any network it may potentially connect to; clearly this would not be the case for a wireless mobile node. To address this the DAR extension to SCTP was developed [16].

DAR allows addresses to be dynamically added and deleted from an association. The DAR extension defines a new message type called an Address Configuration Change (ASCONF). The ASCONF message contains an IP address and a parameter specifying whether the IP is to be added, deleted or changed to be the primary.[3] This message can be transmitted by either endpoint to inform its peer of IP addresses through which it is currently reachable and can be done dynamically during an active association. It is this feature of SCTP that is critical to support seamless handovers.

3.1.2. Partial reliability extension

Although basic SCTP without any extensions can support unordered message delivery, it is not suitable for the transport of real time applications such as VoIP. This is due to the TCP based retransmission mechanisms which can lead to head of line blocking resulting in unacceptable delays. The Partial Reliable SCTP (PR-SCTP) [17] extension to SCTP eliminates this problem allowing SCTP to provide varying levels of reliability to upper layer protocols.

To achieve this PR-SCTP allows the sender to specify a time-to-live parameter on a per message basis. The time-to-live parameter defines the duration for which the sender should attempt to transmit the message. If this parameter expires before the message has been acknowledged SCTP drops the message and does not attempt any retransmission. Essentially, PR-SCTP allows the sender to define how persistent the transport layer will be at attempting to deliver each message.

[3] The *Set Primary* parameter of an ASCONF message is used to instruct an endpoint to begin using a specific alternate address defined in the association. This is different from the SETPRIMARY command used in SCTPs socket API.

3.2. SIGMA – SCTP based handover

SIGMA [1] is a transport layer mobility mechanism based on SCTP, similar to MSCTP. It was designed to be an end-to-end handover solution which does not require any infrastructure support. SIGMA utilises IP diversity and the DAR extension to SCTP to perform seamless handovers for mobile hosts between wireless networks.

SIGMA comprises of the following four steps.

1. *Acquire new IP address* – when the Mobile Host (MH) moves into the coverage area of a wireless access network it is assumed that it can detect the availability of this network. Most wireless technologies have quite sophisticated discovery mechanisms and can detect the presence of a wireless access network if available.
2. *Add IP* – once the MH has acquired a new IP address it must add this to the association by informing the CN of the availability of the new IP address. This is done using DAR.
3. *Handover decision* – the handover decision is based on the RSS of each available AP which is being constantly monitored. When the RSS of the newly available AP becomes greater than that of the existing AP, a handover is triggered. To perform the handover SIGMA must redirect the data flow to the Correspondent Node (CN) via the new AP. The handover is done by sending a 'Set Primary' ASCONF message to the CN containing the new Primary address.
4. *Remove IP address from association* – the final step in the handover process is to remove the old IP address from the association, so that no data is transmitted to the MH via the old access network which is no longer available. Once again, an ASCONF message is transmitted to the CN containing the 'Delete IP' parameter.

A key problem with SIGMA is that it does not consider any QoS parameters when making a handover decision. The work proposed in this paper extends SIGMA by incorporating parameters that directly effect voice quality into the handover decision process.

3.3. The E-Model

The E-Model is a computational model for estimating the subjective quality of a VoIP call, the primary use of which is in the design of codecs and transmission networks. It is standardised by the International Telecommunications Union Technical standards (ITU-T) as G.107 [18].

The E-Model combines loss and delay impairments based on the concept that perceived quality impairments are additive. The output of the E-Model algorithm is a scalar rating of call quality, R:

$$R = Ro - Is - Id - Ie + A \quad (1)$$

where Ro is the Basic signal-to-noise ratio, Is represents impairments simultaneous to voice encoding, Id is impairments due to network transmission, Ie represents effects of equipment and A is the advantage factor. The parameters Ro and Is are associated with the voice signal and therefore are not affected by transmission over the network. The only variable parameters are those affected by network transmission namely Id and Ie. The advantage factor A is used to offset the reduced quality users may be willing to accept in certain circumstances such as in a mobile environment.

The E-Model output R can be converted into the more commonly used voice quality metric MOS using a simple non-linear scaling algorithm defined in G.107. Using this algorithm the relationship between R and MOS as shown in Fig. 1 can be acquired. Table 1 shows the commonly accepted thresholds for user satisfaction levels.

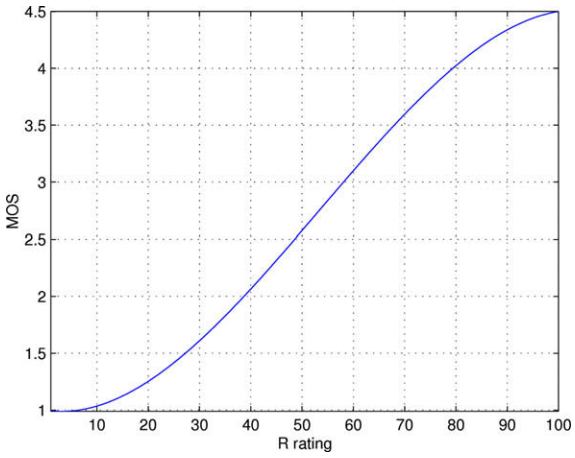

Fig. 1. MOS as a function of *R*.

4. Cross-layer metrics

In order to use the E-Model to calculate accurate estimates of VoIP call quality a number of network metrics that directly effect call quality must be accurately obtained. By using a cross-layer approach ECHO can utilise information from each of the layers in the protocol stack and hence make more informed handover decisions. The approach proposed in this work uses the following parameters from each layer:

- *Physical layer* – depending on the underlying access technology, physical layer parameters such as RSS and Signal to Noise Ratio (SNR) are monitored. This enables the MN to predict when a network will become unavailable and offers the ability to compare the physical layer parameters from each of the available access networks. Also, the physical layer is used to proactively scan for available access networks.
- *Network layer* – on detection of a newly available access network, the network layer is used to obtain an IP address on that network.
- *Transport layer* – once a new IP address is obtained SCTP adds the new IP to the association. Round Trip Time (RTT) metrics for that network can then be obtained at the transport layer.
- *Application layer* – the application layer is used to calculate loss and jitter values for each available access network.

4.1. Physical layer parameters

The proposed handover scheme continually monitors the RSS of each IEEE 802.11 access network to which it is connected. Since the work was implemented in a Linux environment it was possible to use the wireless tools extension to Linux available from [19]. Wireless tools provides extensions to the Linux kernel which allow a user to manipulate the wireless card in a uniform way.

Table 1
G.107 VoIP call ratings.

R value (lower limit)	MOS (lower limit)	User perception
90	4.34	Very satisfied
80	4.03	Satisfied
70	3.6	Some users dissatisfied
60	3.1	Many users dissatisfied
50	2.58	Nearly all users dissatisfied

Wireless tools has three individual modules which are dependent on one another. A user interface provides standardised commands for a user to manipulate the extensions, a modification to the Linux kernel to support the extensions and a hardware interface implemented in each card driver which maps the extensions to the individual hardware implementations [19].

To allow the handover decision function to monitor the parameters available through wireless tools an interface between the handover decision function and wireless tools was required. The developed code allows the handover decision function to query wireless tools to obtain information about a specific interface. The following parameters can be queried and used by the handover decision function:

- Obtain AP Media Access Control (MAC) Address.
- Obtain AP Extended Service Set Identifier (ESSID).
- Obtain Signal Strength from AP.
- Obtain Link Quality.
- Obtain Noise Level.

By using these parameters the handover decision function can have detailed knowledge of the radio environment and the available 802.11 access networks. In most cases the RSS is used to select the optimal candidate handover network and to estimate when network coverage will be lost. However, simply basing a handover decision on physical layer parameters can lead to poor handover decisions as will be discussed later in Section 7.2 and hence information from other layers is also required.

4.2. Network layer parameters

Once a new network is detected at the physical layer the MN must associate with the new network and obtain an IP address. For this work it is assumed that the MN can connect and obtain an IP address using Dynamic Host Configuration Protocol (DHCP).

Having obtained the new IP address it must be added to the existing SCTP association. The proposed scheme gets the IP address from the network layer and adds it to the association using the address reconfiguration extension to SCTP.

4.3. Transport layer parameters

SCTP maintains individual parameters for each IP address in an association which can be used to assess the quality and reliability of the network link. Specifically, this work uses the Smoothed RTT (SRTT) value calculated by SCTP as the delay parameter required for voice quality assessment.

Each link is monitored using the acknowledgements for both heartbeat packets[4] and data packets. Each time data is transmitted it must be acknowledged with a Selective Acknowledgement (SACK) chunk. Based on these, a measure of the RTT for each link can be obtained. A SACK chunk must be sent for at least every second packet received and should be sent within 200 ms of the data chunk which it is acknowledging having been received. Unfortunately with a delay of up to 200 ms before a SACK is transmitted RTT calculations can potentially be incorrect and have high amounts of variability.

SACK chunks can be bundled with data being transmitted from the receiver to sender. Since full duplex VoIP calls are being used, each endpoint will be transmitting packets at short intervals. As every second packet received must be acknowledged this greatly reduces any potential delay in the SACK chunk.

[4] Heartbeat packets are used in SCTP to know which of the IP addresses defined in the association are currently reachable.

Another hindrance to accurately estimating the RTT is *Nagle's algorithm* [20], used by default in SCTP. Nagle's algorithm was developed to improve the efficiency of TCP by reducing the number of packets being transmitted across the network. It is particularly useful for increasing efficiency of applications which regularly transmit small packets. Small packets have huge overhead due to the relative size of the header required compared to the actual payload. Nagle's algorithm works by combining multiple small packets into a single outgoing message thereby reducing the overhead per packet.

The algorithm continually buffers outgoing packets often until it has filled an outgoing packet to the Maximum Transmission (MTU) size. For real time applications such as VoIP, this can have significant problems. Essentially Nagle's algorithm increases the per packet delay and increases the delay variability. Depending on the codec being used and the network delay this can be detrimental to call quality. Further, the algorithm will also delay SACK chunks being transmitted and hence greatly affect the accuracy of the RTT values. It is for these reasons that Nagle's algorithm is disabled in this work.

The combination of packets being transmitted regularly at short intervals and disabling Nagle's algorithm allows accurate RTT estimates to be achieved by SCTP. SCTP uses the same *Van Jacobson algorithm* for maintaining RTT estimates as TCP [21]. When a data packet is transmitted a timer is started, this timer serves the dual purpose of timing how long it takes to be acknowledged and to trigger a retransmission if no acknowledgement is received. Based on the time taken to receive an acknowledgement, M, the RTT is updated according to the formula:

$$\mathrm{RTT} = \alpha \mathrm{RTT} + (1-\alpha)M \tag{2}$$

The weighting factor α defines how much weight is placed on the previous RTT estimation. In all standard implementations, typically $\alpha = 7/8$. This essentially creates a kind of moving average of RTT giving some immunity to variability of the individual RTT measurements.

This estimation of the RTT value is used as the delay measurement parameter for the E-Model. In an 802.11 wireless network the downlink is usually the bottleneck and during congestion has much greater delay values than the uplink; hence most of the estimated RTT time consists of the one-way downlink delay. Considering the RTT as opposed to assuming network symmetry and estimating one-way delay as half the RTT gives a more conservative estimation and hence leads to a more reliable handover decision.

To show that this approach gives accurate representation of the true network delay, the accuracy of the SCTP RTT values was analysed. An experiment was setup using Netem[5] to emulate network delay; this is then compared to the delay estimated by the SCTP RTT values. The results of these experiments are shown in Fig. 2. As can be seen there is a high correlation between the estimated delay and the actual values set using Netem.

4.4. Application layer parameters

Two parameters are measured at the application layer, packet loss and jitter. Each of these will be discussed separately, however SCTP parameters which affect the calculation of loss and jitter will be briefly discussed first.

Normal SCTP provides reliable end-to-end communication with strict in order message delivery on a per stream basis. As is well known, in order delivery can cause head of line blocking resulting in significant delays which in turn increases jitter, making normal SCTP unsuitable for the transport of real time applications. The partial reliability extension to SCTP addresses these issues and hence is used in this work.

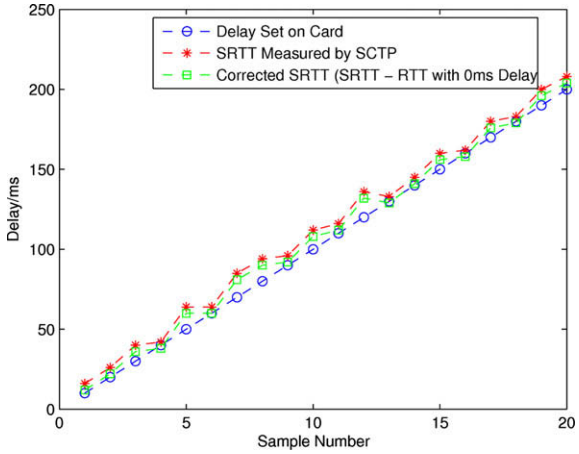

Fig. 2. Accuracy of SCTP estimated delay metrics.

As discussed in Section 3.1.2 the PR-SCTP extension allows SCTP to provide User Datagram Prototcol (UDP)-like packet delivery while preserving the TCP friendly congestion control mechanisms of SCTP. This is achieved by assigning a Time to Live (TTL) parameter for each message being transmitted. If this timer expires before the message has been successfully transmitted, the message is dropped. This allows SCTP to provide a more VoIP friendly transport mechanism as individual lost packets will not have as significant an impact on overall quality.

Selecting an optimum TTL value is critical to the performance of VoIP over SCTP. If the value is too low, packets will be dropped even though they could have potentially still been used by the VoIP application. However, a value which is too high will lead to periods of head of line blocking thereby causing increases in delay. As low delay values below 150 ms have no affect on call quality the selected TTL value must not be below this. Also, delay values above 400 ms greatly reduce the interactivity of voice calls essentially leading to half duplex communication. A TTL value of 500 ms was selected as this will allow packet delays up to 500 ms slightly above the 400 ms threshold. Delayed packet transmissions above 500 ms will be dropped and will therefore be considered lost.

4.4.1. Calculating jitter

Packet jitter is calculated at the application layer based on the downstream packets being received. Each time a VoIP packet is transmitted the local system time is encapsulated within the packet. At the receiver these timestamps can used to calculate the interarrival jitter. For example, when using the G.711 VoIP codec with a frame size of 10 ms, the receiver would expect to receive a packet every 10 ms in the absence of jitter. Therefore any variability in this interarrival delay can be used to calculate a jitter value.

Jitter is defined as the difference between the *relative transmit time* between two packets. Where the *relative transmit time* is the difference between the transmit timestamp of a packet and the system time at the receiver when the packet is received. The interarrival jitter is a smoothed value of the jitter values being obtained.

Jitter is calculated using the E-Model recommended Real-Time Transport Protocol (RTP) jitter algorithm from RFC1889 [23]. Since the interarrival jitter calculation is based on previous jitter values and as the initial timestamp and the local system time will have random values, the initial jitter calculation will be incorrect. As a smoothed value is being considered this initial incorrect value will impact the interarrival jitter being calculated over a period much greater than the duration of two packets. Although the incorrect value will be filtered out as packets continue to be received and the

[5] Netem provides network emulation for testing protocols. It is already enabled in most recent Linux distributions [22].

jitter calculation is updated; it means that any QoS calculation being made during the initial period of transmission will be incorrect.

To account for this no jitter calculation is done for the first packet received on any link. Rather the jitter value is initialised to zero and the timestamps are recorded for the next time jitter will be calculated. This solves the problem of having initial random timestamp values causing incorrect jitter calculations. The function for calculating the interarrival jitter is shown in Fig. 3.

4.4.2. Calculating loss

As with jitter, packet loss is calculated using the individual downlink VoIP streams over each access network. Each VoIP packet contains a sequence number that is used to monitor which packets have been lost. As stated previously any packet transmission delayed for longer than 500 ms will result in the packet being dropped and considered lost. Hence, the recorded packet loss rate is the result of both these timeouts and losses within the network.

Each time a packet is received its sequence number is checked, and any packet up to the current sequence number which has not been received is noted. The packet loss rate is then calculated as a moving average. A minimum moving average window size of 100 packets is needed to obtain a resolution of 1% as is required to relate the loss rate to a quality degradation in the E-Model [24]. For this reason during the initial 100 packets being received over a network link the loss rate resolution does not have an accuracy of 1%. For example, if the G.711 VoIP codec with a frame size of 10 ms is being used it takes 1 s before the 1% resolution is achieved. In this work, a window size of 200 packets is used to obtain greater resolution and achieve more averaged value to compensate for any sudden burst of packet loss.

As was done with the RTT metrics, the accuracy of the loss metrics was analysed using Netem to emulate packet loss. This is then compared to the loss estimated by the application layer mechanism. The results of these experiments are shown in Fig. 4. As can be seen there is a high correlation between the estimated packet loss and the loss rate set using Netem.

5. Endpoint centric handover – ECHO

This section gives an overview of the ECHO mechanism. ECHO is an end-to-end handover mechanism which uses SCTP to transport VoIP data, essentially extending the SIGMA approach to mobility. It should be noted that although ECHO currently focuses on VoIP it can be extended to provide QoS aware handovers for any streaming application. This assumes that accurate mechanisms for measuring the quality of the media can be obtained. Indeed previous work has already shown the ability of SIGMA to provide seamless handover of video streams [25]. ECHO is optimised for the transport of VoIP data by using the partial reliability extension and unordered packet delivery of SCTP. This allows SCTP to achieve similar performance to that of UDP while maintaining the multihoming capability and congestion control mechanisms. By leveraging the multihoming capability of SCTP, ECHO can affect in call seamless handover of VoIP without any degradation in call quality. ECHO is made up of four main discrete components as shown in Fig. 5.

```
void calc_jitter(long arrival, long ts, call_params *link){
//ts = timestamp of received packet
//arrival = local system time when packet was received
long transit = arrival - ts;
long d = transit - link->transit;
link->transit = transit;

if(link->first_packet_rxed < 2){
link->jitter = 0.0;
d= 0.0;
link->transit = transit;
link->first_packet_rxed++;
}
else{
if (d < 0) d = -d;
double d_temp = (double)d;
d_temp = d_temp/1000.0; //Convert from usec to ms
link->jitter += (1./16.) * (d_temp - link->jitter);
}
}
```

Fig. 3. Function to calculate interarrival jitter.

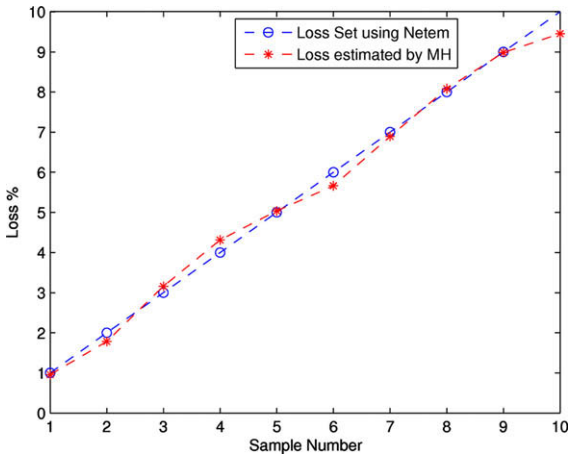

Fig. 4. Accuracy of estimated packet loss rate.

Each of these components performs the following particular tasks:

- *VoIP call handler* – this component is responsible for initiating and terminating the full duplex VoIP calls between the MN and CN. Currently the call handler supports both the G.711 and G.729 VoIP codecs. The downlink VoIP stream at the MN is monitored and each time a packet is received it is stored in the *Parameter Store* and a message is sent to the handover decision function informing it of the newly received packet.
- *Parameter store* – the zparameter store component is simply a structure which stores all information related to each link the MN may have. These parameters include but are not limited to RSS, Loss, Jitter, RTT and MOS. The packet store is used to store up to date parameters used by the *Handover decision function* to enable QoS aware handover decisions.

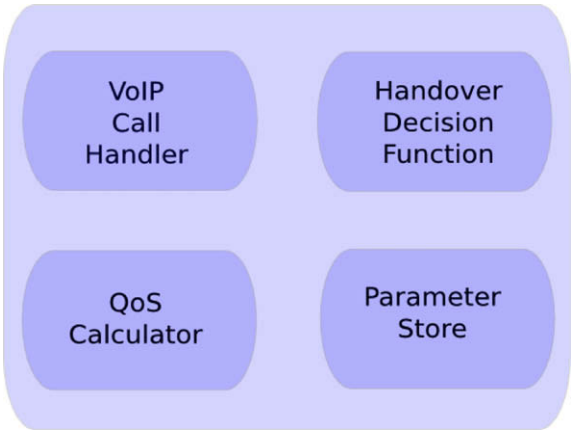

Fig. 5. ECHO components.

- *QoS calculator* – this component is made up of multiple algorithms for calculating the required QoS parameters. Specifically, it incorporates a loss calculation function, a jitter calculation function and a real time implementation of the E-Model algorithm. The results from these calculations are stored back into the *Parameter Store*.
- *Handover decision function* – this component is the most complex element of ECHO. It continually monitors both the VoIP call QoS and RSS parameters from each access network in order to determine when a handover decision should be made. To accomplish this the packet duplication functionality is implemented in this component. Further details relating the handover decision will be discussed later in Section 5.2.

5.1. ECHO initialisation

The ECHO initialisation process consists of the following three steps:

- Establish VoIP call over PR-SCTP and Start the ECHO Daemon.
- Begin monitoring the downlink VoIP stream.
- Begin monitoring the physical layer parameters from each available Network Interface Card (NIC).

The ECHO daemon is created once a VoIP call is established between two ECHO supporting nodes. Currently, the VoIP call must be full duplex, using the G.711 or G.729 VoIP codec, however this could be expanded to include other codecs.

Once the daemon is created ECHO begins to monitor the downlink VoIP stream call quality. When considering 802.11 access networks it is the downlink that usually becomes the bottleneck and it is for this reason that ECHO focuses on the downlink VoIP stream. Also, it is worth noting that since both endpoints support ECHO each will be monitoring their own downlink and this will account for the full duplex call.

The metrics from the downlink VoIP call, specifically RTT, jitter and loss as previously discussed, are mapped to a MOS score in real time. This allows ECHO to continually monitor the quality of the ongoing VoIP call. Although not yet implemented, this feature could be used to allow ECHO to dynamically modify the PR-SCTP parameters based on the VoIP call quality. For example, if packet loss begins to occur while the RTT is relatively low, the TTL parameter can be increased thereby decreasing the amount of packet loss at the expense of increased delay.

If the primary interface being used is an 802.11 wireless network, ECHO must also monitor the RSS from the AP through which the MN is currently communicating. Given that the MN may only be single homed, it must locate other access networks as possible candidate handover networks. This is a problem as it requires the presence of a second 802.11 NIC in order to scan for other 802.11 networks. In the proposed work, a second card is required, however some work has been done, which in the future could be leveraged by ECHO to remove this requirement.

Microsoft have proposed a technology called *Virtual WiFi* [26] which allows a single card to simultaneously connect to multiple access networks. A layer which provides the virtualisation and mapping is introduced between the IP layer and the physical layer. Currently, switching delays between networks takes from 100 to 600 ms, however by using modified cards delays values as low as a few milliseconds can be obtained. Work is continuing and it is believed that with further modifications delays in the order of 100 μs can be achieved [27].

Using the second card a MN has the ability to scan for other available access networks without affecting the ongoing call over the primary interface. The scanning feature of the wireless tools module is used to continually scan for access networks different from the access network being used as the current primary interface. The MAC address of the current primary AP is compared to the MAC address of APs being picked up by the second card to prevent it from connecting to the same AP as the primary. Having detected a new access network the second card attempts to associate and obtain a new IP address. This work assumes that the available network is open and does not require authentication, or that the MN can automatically authenticate without requiring any user interaction.

It is also assumed that if other networks are available such as an 802.16 network that it is an overlay network which is consistently available. Hence, there is no requirement to monitor any physical layer parameters as with 802.11 networks. This assumption is made in the implementation due to the available technology and will be discussed further in Section 6. It should be noted that as technologies such as mobile WiMax (802.16e) [28] become available, physical layer parameter monitoring of these access networks can be introduced into ECHO using a similar approach to that used for 802.11.

In the case of 802.11 networks ECHO uses an RSS threshold to trigger a handover decision. Calculation of the predefined threshold is based on multiple experiments details of which will be discussed in Section 7.3.2.

5.2. Handover decision function

When the MN detects multiple available networks it is in the overlapping coverage region of two or more access networks and ECHO will begin to monitor the RSS of each AP. When the RSS of the new access network becomes greater than that of the existing access network, the handover decision process is triggered. It is at this point that ECHO differs from other mobility mechanisms such as SIGMA and MSCTP. These other schemes would have immediately performed a handover to the new access network based on the RSS values without considering the QoS that will be obtained after handoff. ECHO on the other hand, bases the handover decision on both the RSS and the available downlink QoS from each access network.

5.2.1. Packet duplication

In order to calculate the downlink QoS for each link independently, the MN must be receiving data over both access networks. To accomplish this, the MN transmits a message to the CN informing it to begin duplicating the downlink data over both networks simultaneously.

Packet duplication messages are transmitted using modified RTP packets. Additional RTP payload types were created, these are *SCTP_START_DUPLICATION* and *SCTP_STOP_DUPLICATION*. As only the header information is required, each packet duplication message contains no data and hence has very little overhead.

The multistreaming feature of SCTP is used to prevent the ongoing VoIP call from interfering with the timely delivery of the packet duplication control message. Multistreaming allows the control messages to be transmitted and received independent of one another thereby preventing any delay in delivery due to head of line blocking. To decide which stream to use ECHO checks the stream ID being used for the VoIP call and uses the next available stream to transmit and receive the control messages. This guarantees that both the sender and receiver are using the same stream for control messages.

Since the partial reliability extension is being used for transporting the VoIP data any packet whose transmission is delayed for an extended period of time will be dropped and therefore lost. Clearly this behaviour is unacceptable for control messages for which delivery is imperative. To avoid this problem different timeout values are set when transmitting a control message. Setting the

timeout value to a value called *SCTP_LIFETIME_RELIABLE*, essentially sets the time-to-live parameter to 0 specifying that no timeout will occur for that message.

On successful reception of the *SCTP_START_DUPLICATION* message, the CN will begin transmitting the same downlink VoIP stream to each of the MN IP addresses specified in the association. Since each IP address is specific to each access network this has the effect of simultaneously transmitting duplicate data over each of the access networks being used by the MN. Although this approach will increase the traffic load on the uplink of the CN, any congestion caused will effect both streams to the same extent and hence still allow a comparison between access networks to be made. In order to allow ECHO to operate with both MN and CN being mobile, further work would be needed to investigate the behaviour when each node has multiple networks available.

As with the control messages, independent SCTP streams are used for each network over which the duplicate data is being transmitted. Since the duplicate stream is being used to assess the QoS that can be achieved over the alternate network it must not be interfered with by the delivery of data over the primary interface. Using different streams for the normal VoIP data and the duplicate data prevents any potential delays or head of line blocking. Once the MN has assessed the QoS of the alternate network it transmits an *SCTP_STOP_DUPLICATION* control message to the CN. When this is received by the CN it stops duplicating data over the alternate path and resorts back to using only the primary interface.

Although this approach is transmitting duplicate data which may not be have any benefit, immediate benefit can be obtained under certain circumstances. During periods of packet duplication, multiple copies of the same packet will be received by the MN. The MN can then choose to use the packet which arrives earliest. This will provide improved performance even before a handover takes place. It is especially beneficial for environments in which there is poor performance on the primary network, possibly due to the MN being at the edge of the network coverage.

There are however some problems with simultaneously transmitting data across multiple paths within the same association. For example, if a message fails to be successfully transmitted to a particular destination IP address specified in the association, SCTP will attempt a retransmission across an alternate address. This feature gives a high level of reliability and was the main motivation for including multihoming capability in SCTP. Unfortunately SCTP was not initially designed to support concurrent multipath transfer and hence can create problems when it is attempted.

To solve these problems several concurrent multipath transfer algorithms have been proposed by the initial designers of SCTP [29,30]. Once this work is included in SCTP implementations, ECHO will be capable of leveraging the algorithms to implement duplication of packets over all available paths. However, since concurrent multipath transfer is not yet implemented an alternative approach was used. In the implementation of ECHO for this work the problems associated with multipath transfer were addressed by modifying the routing tables of the MN and CN.

5.2.2. Overview of handover decision

A flow chart of the handover process which gives more detail is shown in Fig. 6. As can be seen, ECHO assesses the QoS of the new network before making the handover decision. As discussed earlier the MN transmits a *SCTP_START_DUPLICATION* message to the CN. On reception of this, the CN begins simultaneously transmitting all data to each of the MN IP addresses specified in the association. This allows the MN to individually measure the required network metrics which are then converted to a MOS score using the E-Model algorithm.

The MOS score obtained from the E-Model algorithm is then compared to a MOS score threshold to decide if the call can be supported on the newly available access network. An appropriate MOS threshold was chosen from Table 1. The handover decision process leads to the following two scenarios.

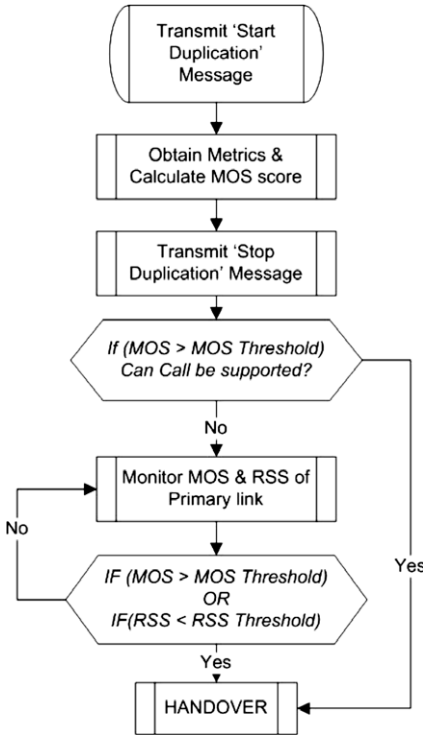

Fig. 6. ECHO flowchart.

- *Scenario 1*– the call can be supported by the new access network. Since the call can be supported a handover is immediately performed.
- *Scenario 2* – the call cannot be supported by the new access network. In this situation, the MN assesses the MOS score being obtained over the existing connection; if the call quality is above the MOS threshold then no handover takes place. The MN will then continue to monitor both the RSS of the existing network connection and the MOS score of the ongoing VoIP call and a handover is triggered if the RSS falls below an RSS threshold. Although the VoIP call may not be supported by the network handing over prevents complete loss of coverage. The value chosen for the RSS threshold parameter is based upon multiple experiments and will be discussed in detail in Section 7.3.2.

5.3. Network metrics

During a packet duplication period the MN uses the downlink traffic streams to obtain network metrics independently for each access network. As previously discussed in Section 4, ECHO must use multiple metrics obtained from different layers of the stack to accurately estimate the MOS value. More specifically, jitter and loss metrics are calculated at the application layer and delay metrics are obtained from the transport layer.

When a new network is added to an association ECHO creates a packet store and a state information store for the newly added network. The packet store essentially acts as a buffer with a size of 200 VoIP packets to store all packets received over that network and the state information store is used to contain up to date information about the interface such as packet loss, jitter, RTT and other state information.

Each time a packet is received over a particular network it is stored in the packet store for that network. The loss, jitter and RTT values for that network are then calculated and updated accordingly. Jitter is calculated using the E-Model recommended RTP jitter algorithm while loss is calculated using a moving average of individual packet losses. The RTT must be obtained from the SCTP RTT estimate for the particular network being examined.

While the MN may have only one destination address specified in the association during handover, the CN will have two or more addresses specified – one IP address for each access network. Current SCTP implementations only retain state information for each active destination specified in the association. Therefore, as the MN may only have one address for the CN specified in the association, it cannot obtain RTT information for both paths. Since the CN will have both interfaces of the MN specified as destination addresses in the association, the RTT for each path must be acquired at the CN. The RTT values obtained at the CN are independently encapsulated into the downlink traffic over each link and transmitted to the MN over both access networks. Having obtained all of the required metrics, ECHO uses the E-Model in real time to calculate a MOS score for each of the candidate handover networks.

5.4. Real time E-Model

The E-Model was primarily designed as a network planning tool and not as a tool for the real time estimation of call quality or for use in live networks. Despite this, the E-Model is being increasingly used as a tool for live speech quality measurement. For example, many popular VoIP call quality assessment tools such as http://www.testyourvoip.com use the E-Model to calculate a MOS score in real time.

In order to successfully use the E-Model in real time some of the required parameters which cannot be obtained through passive measurement must be estimated. To this end work has been done on providing estimates for parameters which do not vary due to network conditions and therefore allow the E-Model to provide valid results when used in real time.

In [31] specific values were chosen for the non-varying parameters within the E-Model. This simplified variant can then be used in a real time context. Based on this, the E-Model can be reduced to the following expression:

$$R = 93.34 - Id - Ie - A \tag{3}$$

Id takes into account network delay and jitter parameters while the equipment impairment factor *Ie* is loss and codec dependent. Another element of jitter which must be considered is the impact that the jitter buffer can have on increasing delay and smoothing out delay variation. This is accounted for by the parameter *Dj* which is an element of *Id* and is dependent on the type of jitter buffer being used. Since most VoIP systems implement a dynamic jitter buffer, *Dj* is estimated using a simple formula which models a dynamic jitter buffer shown in [31]. It assumes a dynamic jitter buffer which increases in size as the amount of packet jitter increases and places an upper limit of 300 ms on the buffer size. This emulates quite a simplistic buffer which would be the minimum implemented by a VoIP system and therefore gives the worst case performance that could be expected. It should be noted that more complex jitter buffers may give higher performance and can be accounted for in the E-Model by modifying the *Dj* parameter.

$$Dj = \min(\text{codec frame size} + 0.9x \text{ RTP jitter}, 300) \tag{4}$$

The advantage factor *A* accounts for a users willingness to accept lower call quality for the ability to make a call. For example a user will usually rate a call in a mobile environment higher than a call of equal quality made from a fixed location. Although the ECHO scheme is operating in a mobile environment an advantage factor value of 0 was chosen. A value of 0 is the same as is used for a fixed location. This is done so that the MOS values calculated can be directly compared with those obtained by many of the available tools. This further reduces the E-Model algorithm to:

$$R = 93.34 - Id - Ie \tag{5}$$

6. Experimental testbed

This section gives an overview of the network topology and architecture of the ECHO testbed used to test ECHO. A description of the scenarios in which ECHO can operate is also presented.

6.1. Testbed architecture

Two different network architectures were developed to test ECHO. A homogeneous network setup consisting of two overlapping 802.11b Wireless Local Area Network (WLAN)s and a heterogeneous network setup consisting of two non-overlapping 802.11b WLANs and a WiMax overlay network

The first to be setup was the homogeneous network as is shown in Fig. 7. This testbed consists of two overlapping 802.11b WLAN access points, two desktop PCs to act as gateways between the APs and an ethernet Local Area Network (LAN). Also included is the MN and a CN; these are the two endpoints between which handover takes place. The MN and CN were implemented on laptop computers running Ubuntu 7.10 (Gutsy Gibbon) with LKSCTP installed. The MN is multihomed having two WLAN cards to allow simultaneous connection to multiple APs as required by ECHO. The CN is single homed and connects directly to the Ethernet LAN with all routing carried out by the two gateway machines.

The heterogeneous network setup is shown in Fig. 8. As can be seen the WiMax network is assumed to be an always available overlay network. This requires that the MN has an extra IP interface capable of connecting to the WiMax network.

6.2. Multiple scenarios

The behaviour of the ECHO mechanism depends upon what networks are available and through which network the MN is

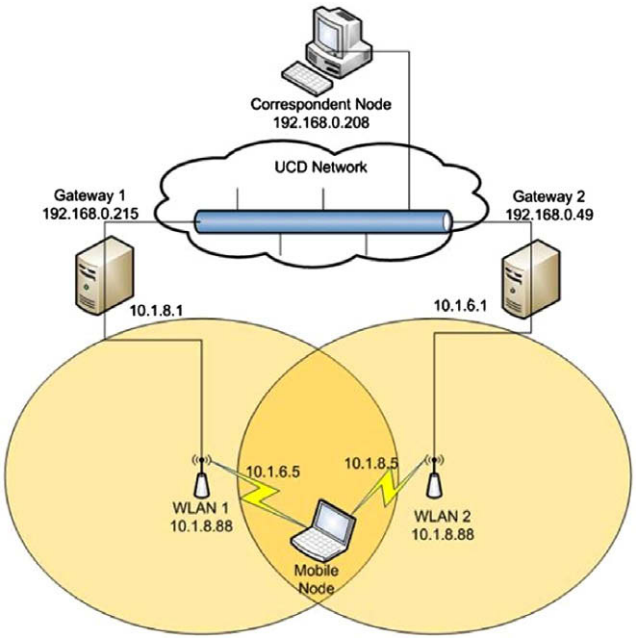

Fig. 7. Homogeneous network topology.

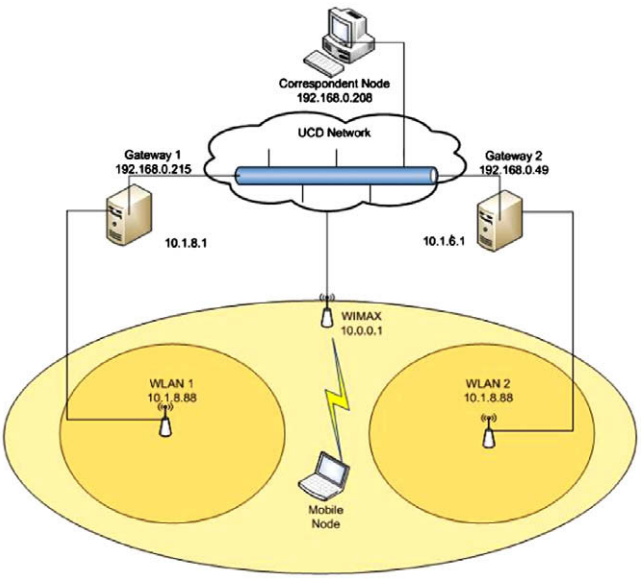

Fig. 8. Heterogeneous network topology.

currently connected. Because ECHO can operate in a heterogeneous network environment consisting of different radio access technologies, a direct comparison of RSS values cannot be used to trigger a handover decision.

The current implementation of ECHO has been designed to operate with 802.11 WLANs and a WiMax network which is assumed to be a ubiquitously available overlay network. For ECHO to operate between these networks ECHO must be capable of making scenario dependent handover decisions. For example, the behaviour required when handing between two WLANs is different to that required for handing over between WLAN and WiMax. The homogeneous case of WLAN to WLAN has already been explained in detail as the example considered in Section 5 and hence will not be discussed further.

For the heterogeneous case it is assumed that using a WLAN is preferable to using a WiMax network due to lower power requirements and that WLANs are more likely to have a lower economic cost. ECHO therefore strives to use a WLAN when one of sufficient quality is available. The following gives a brief description of the handover process for two different heterogeneous scenarios.

- WLAN to WiMax – in this case, the WLAN is being used as the primary network and a handover will only occur if either the MOS or RSS threshold is reached. If the WLAN becomes congested due to an increase in the number of users or if existing users begin to increase the load being placed on the network, the MOS score will be affected. Once the MOS score falls below a predefined threshold ECHO will initiate a handover to the WiMax network. Alternatively, if the MN begins to move out of the coverage area of the WLAN the RSS will decrease. If the RSS threshold is reached, ECHO will initiate a handover to the WiMax network assuming no other WLAN is available.
- WiMax to WLAN – in this case, the WiMax network is being used as the primary network due to the unavailability of a WLAN capable of meeting the required QoS. As the MN moves, a new WLAN may be detected, ECHO immediately begins monitoring the RSS of the new network. If the MN continues to move closer to the new network AP the RSS will increase. Once the RSS threshold is reached ECHO initiates packet duplication to assess the call quality over the new network; if the VoIP call can be supported ECHO performs a handover to the WLAN.

7. Results

This section describes results demonstrating a real implementation of the ECHO handover mechanism. First, results are presented to show the inability of congested WLANs to support VoIP calls, these experiments were carried out on a live WLAN network on campus in University College Dublin (UCD). This is followed by results which demonstrate the poor handover decisions that can be made by existing RSS based handover solutions. Example results using the ECHO mechanism in multiple scenarios are then presented. Due to space constraints only a limited number of results are presented, however these results show the ability of ECHO to achieve seamless QoS based handovers in a number of different scenarios.

7.1. VoIP call support in live WLAN networks

Many existing WLAN networks use older 802.11b APs and the rapid growth and popularity of WLAN capable devices has begun to put a strain on these networks. Although using the newer 802.11g APs can alleviate this problem somewhat, the greatly increased performance is only achieved when the MN is relatively close to the AP. Also, with the increasing number of devices it is only a matter of time before these networks also become congested. Besides, the existing 802.11b APs will continue to be used for the foreseeable future.

To demonstrate the inability of congested WLANs to support VoIP calls a number of experiments were performed. These experiments focused on 802.11b APs that are in popular locations on UCD campus and hence have high volumes of traffic. Specifically, the experiments focused on two APs that are located in study areas of the business school in UCD and were performed at times of peak traffic.

In these experiments, half duplex VoIP calls were established between two nodes connected to the same AP using the client/server software previously described. Half duplex calls were used to focus on the downlink VoIP traffic only, since in a normal scenario the call would not be terminated on the same AP from where it originated.

Fig. 9 shows the downlink MOS of a VoIP call over the live WLAN network. As can be seen the call quality is consistently low and the network cannot realistically support the call.

The primary reason for the low call quality is packet loss. As can be seen in Fig. 10 the packet loss levels fluctuate and at many times have very high values of over 15%. This level of packet loss makes all VoIP calls impossible regardless of the codec being used. In these experiments the G.711 codec was used; this codec has no

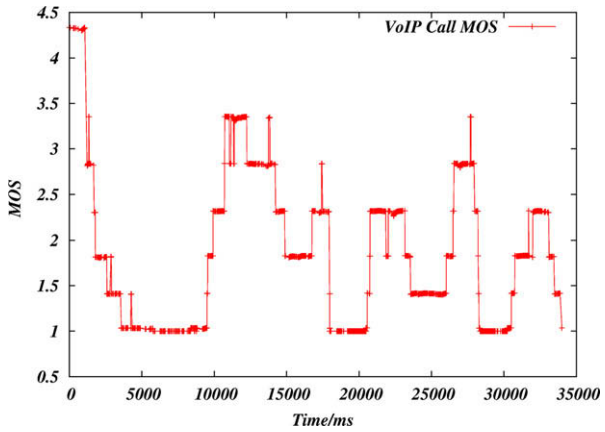

Fig. 9. VoIP MOS on congested live WLAN.

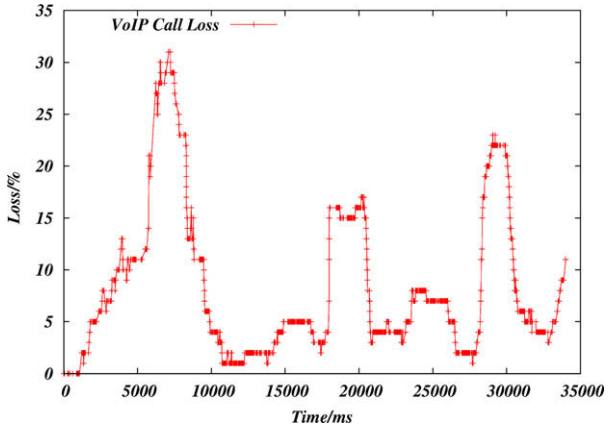

Fig. 10. VoIP loss on congested live WLAN.

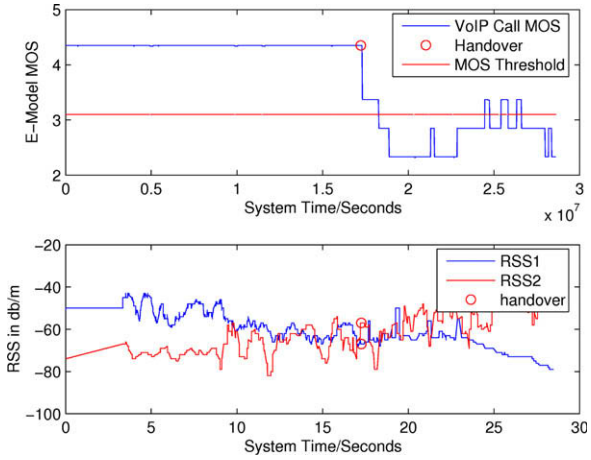

Fig. 11. RSS based SCTP handover to network with high packet loss.

inter-packet dependencies and hence suffers less in the presence of loss than lower bit rate codecs such as G.729. Therefore using lower bit rate codecs would further degrade the call MOS value.

7.2. RSS based handover schemes

Existing SCTP based handover mechanisms such as SIGMA and MSCTP do not consider any QoS parameters when making handover decisions. Only considering simple physical layer metrics such as RSS and SNR can lead to these schemes making poor handover decisions [2].

An experiment was setup to demonstrate cases in which only considering RSS leads to poor handover decisions. The testbed shown in Fig. 7 was used to perform the experiments. The initial primary AP was uncongested, however the secondary AP to which handover will take place was highly congested and experiencing high packet loss.

The results of this experiment are shown in Fig. 11. The voice call quality measured in MOS is initially at the maximum attainable value of 4.4. As in the previous experiment the MN moves at walking pace from the coverage area of AP1 towards AP2. As the MN moves further into the coverage area of the second AP, the RSS from AP2 becomes greater than that of AP1, at which point a handover is performed to AP2; this occurs at approximately 17 s. Since AP2 was highly congested, the call quality immediately dropped to an unacceptable level. In this case, although the RSS of the alternative network was higher than that of the primary, it did not give an accurate reflection of the achievable call quality. This demonstrates the requirement to consider multiple metrics when making handover decisions.

7.3. Performance of ECHO handover scheme

In this section, results are presented using the ECHO mechanism. By incorporating multiple metrics into the handover decision, ECHO can achieve better performance than RSS based approaches which only consider a single physical layer metric. Unlike the RSS based approaches ECHO assesses metrics that directly effect VoIP quality and hence can maintain optimal VoIP call quality. Another major advantage that ECHO has over the RSS based approaches is that it can perform handovers between dissimilar access technologies and hence operate in a heterogeneous network environment.

Each handover experiment was carried out on the testbeds shown in Figs. 7 and 8.[6] As in the previous experiments the MN moved at walking speed which is realistic for the scenarios considered. Higher speed mobility would most probably require the use of longer range radio technologies. It is worth noting that as the MN moves away from each AP standard Link Adaptation (LA) occurs; however, this did not have any significant impact on the results.

7.3.1. QoS based ECHO handover

Fig. 12 shows the result of an experiment for the same scenario as the SIGMA handover shown in the previous experiment, except in this case the ECHO handover mechanism was used. When the RSS of the new network becomes greater than that of the existing network the QoS mechanism assesses the call quality that can be achieved over the newly available network. ECHO estimates that the secondary link will provide a low quality MOS score of approximately 2.3. This score is then compared to a MOS threshold, which in these experiments was chosen to be 3.1. This value was chosen from Table 1 as it is the value below which many users become dissatisfied. Since the calculated MOS score is below the required threshold no handover takes place and the call continues over the existing AP at high quality.

7.3.2. Calculating the RSS threshold

In the previous experiment, no handover took place as the call could not be supported by the secondary network. However, if the MN continues to move away from the primary AP the RSS will continue to decrease, eventually packet loss will begin and

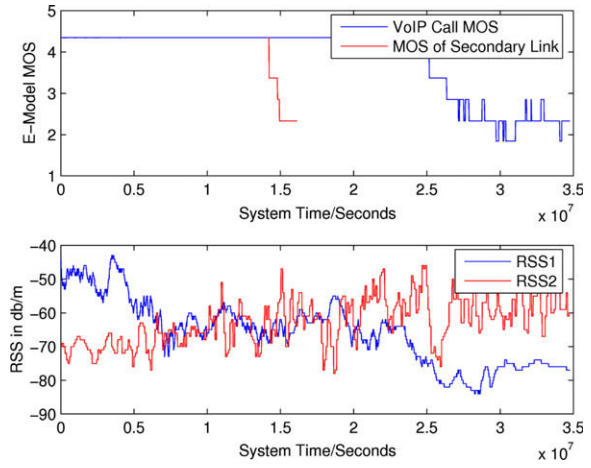

Fig. 12. MOS based handover to high loss network – no handover.

[6] In some experiments it was required to emulate congestion on the WLAN APs, this was done by introducing 5% packet loss at the gateway using NETEM.

coverage will be lost. Therefore, the MN must handover prior to moving completely out of the coverage area of the AP. Although, packet loss can give a good indication of being at the edge of a coverage area, it is very difficult to differentiate between this type of packet loss and packet loss due to other causes such as congestion.

In order to estimate when coverage will be lost ECHO uses an RSS threshold, defined as the mean RSS at which packet loss begins to occur. Multiple experiments were performed to calculate the optimum RSS threshold value. Each experiment involved a MN moving at walking pace away from the 802.11b access point, to which it was connected, while measuring the packet loss and RSS. The results of one such experiment are shown in Fig. 13, this shows that when an RSS of −76 dbm is reached, application packet loss begins to occur and rapidly increases. Twenty such experiments were performed and the mean RSS at which packet loss occurs was found to be −78 dbm.

7.3.3. ECHO handover incorporating QoS and RSS capabilities

Incorporating the RSS threshold into ECHO allows connectivity to be maintained when loss of coverage is inevitable. Even though this will result in the call quality being reduced, it is better to maintain connectivity than disconnect completely. Fig. 14 shows an ECHO handover experiment utilising the RSS threshold feature.

As in the previous experiments when RSS1 becomes less than RSS2 the quality of the secondary link is assessed and since it cannot support the call at a high quality no handover takes place. The MN then continues to move away from the primary access point. When the RSS threshold is reached on the primary link a handover is triggered. The handoff is performed before coverage is lost which would prevent the handover from taking place. In this scenario, the QoS handover mechanism achieved a higher quality VoIP call for 11 s longer than would have been achieved if an RSS based handover process was used. This shows the ability of ECHO to provide optimal call quality and achieve higher performance than simple RSS based mechanisms such as SIGMA.

7.4. ECHO handover in a heterogeneous network

This experiment demonstrates the ability of ECHO to perform multiple handoffs across different access technologies in a heterogeneous network environment using the testbed topology shown in Fig. 8. Since both networks are of different radio access technologies a direct comparison between the RSS values cannot be made. For this reason, the RSS threshold described earlier is used to detect the coverage edge of the WLAN.

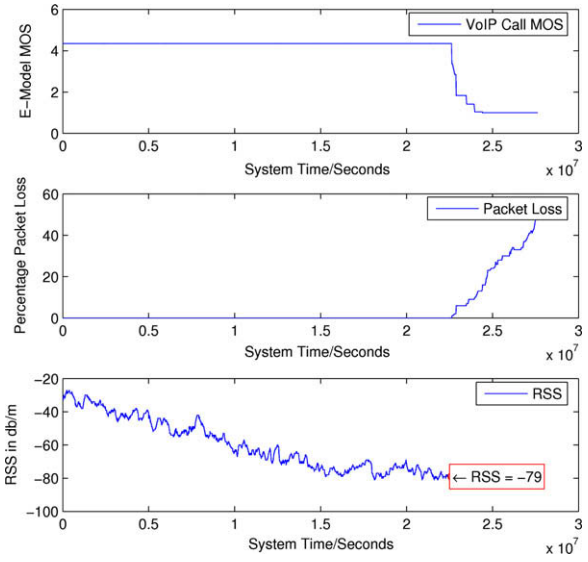

Fig. 13. Determining the RSS threshold.

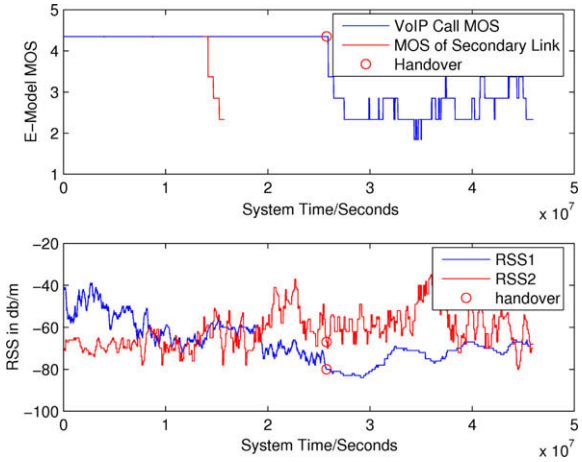

Fig. 14. ECHO handover using QoS and RSS capabilities.

The MN is initially connected to WLAN1 over which a VoIP call is ongoing. The MN then begins to move out of the coverage area of WLAN1 toward WLAN2. In this case, there is no overlapping coverage area between both WLANs and the MN must perform a handoff to the WiMax network. This handoff is performed based on the RSS of WLAN1 reaching the predefined threshold as shown in Fig. 15 at approximately 19 s.

As the MN continues to move toward WLAN2 the network is detected and the RSS monitored. Once the RSS reaches the predefined threshold, ECHO assesses if the call can be supported on WLAN2. As shown in Fig. 16, ECHO accurately estimates the call quality to be approximately 4.336 and since this is above the MOS threshold it is decided that the call can be supported. Hence, a handover to the new WLAN is performed; this occurs at approximately 45 s.

Although no packet loss is experienced during or after handoff, there is a marginal increase in delay due to the increased delay on the WiMax network. This delay is due to the frame size of 20 ms being used by the WiMax equipment and the delay values experienced in these experiments agrees with other previously published work on QoS in 802.16 networks [32]. The increase in delay however is relatively small and has little impact on the call quality as shown in Fig. 16.

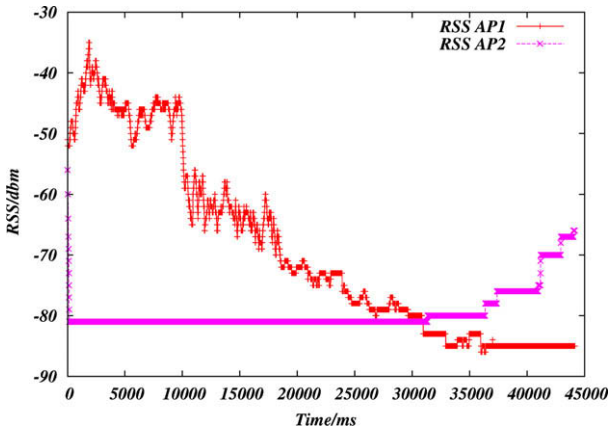

Fig. 15. RSS for WLAN to WiMax to WLAN.

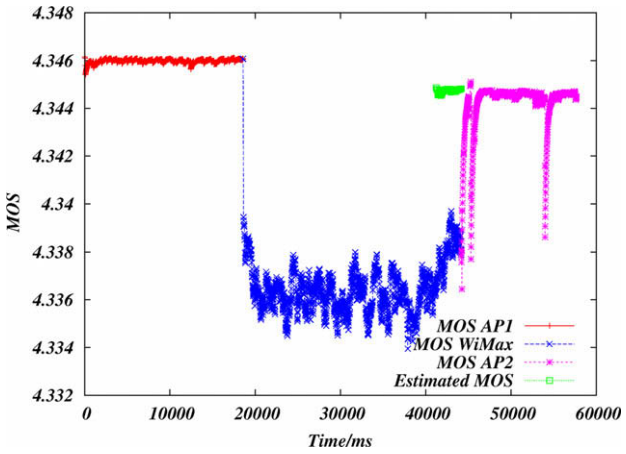

Fig. 16. MOS for WLAN to WiMax to WLAN.

8. Conclusion

In this paper, an endpoint centric handover mechanism for VoIP called ECHO was proposed. ECHO enhances the SIGMA mobility scheme by considering QoS metrics as part of the handover decision process. It was shown that by not considering the QoS that will be obtained after handoff SIGMA can make poor handover decisions. ECHO addresses this by considering multiple cross-layer metrics that directly effect VoIP call quality. The obtained metrics are then mapped to a MOS value using the ITU-T E-Model. ECHO uses MOS scores from each of the candidate handover networks to make QoS aware handover decisions resulting in improved VoIP performance.

Results are presented showing the accuracy of the cross-layer metrics used to calculate the MOS score. The ECHO estimated metrics show a high correlation with the actual values present in the network. An experimental testbed was developed on which to measure the performance of a real implementation of ECHO. Results from experimental evaluation show that ECHO achieves better performance than simple RSS based handover mechanisms. It is also shown that ECHO maintains high VoIP call quality by using the best available access network and minimising non essential handovers. This work has shown that terminal oriented QoS aware handover solutions which do not require network support are possible.

The current implementation of ECHO focuses on providing QoS aware handovers exclusively for VoIP; this was due to the high level of QoS required by VoIP and its sensitivity to network conditions. Future work will investigate extending ECHO to provide QoS aware handover decisions for other streaming media with a particular emphasis on video streaming.

Acknowledgements

The support of the Irish Research Council for Science, Engineering and Technology (IRCSET) is gratefully acknowledged. The work of M. Atiquzzaman was supported by NASA Grant NNX06AE44G.

References

[1] S. Fu, M. Atiquzzaman, Sigma: a transport layer handover protocol for mobile terrestrial and space networks, in: J. Ascenso, L. Vasiu, C. Belo, M. Saramago (Eds.), Invited book chapter in e-Business and Telecommunication Networks, Springer, 2006, pp. 41–52.
[2] J. Fitzpatrick, S. Murphy, M. Atiquzzaman, J. Murphy, Evaluation of voip in a mobile environment using an end-to-end handoff mechanism, Mobile and Wireless Communications Summit (2007) 1–5 (16th IST).
[3] C.E. Perkins, Mobile IP, Communications Magazine IEEE 35 (5) (1997) 84–99.
[4] C.E. Perkins, Mobile networking through Mobile IP, Internet Computing IEEE 2 (1) (1998) 58–69.
[5] D. Johnson, C. Perkins, J. Arkko, Mobility Support in IPv6, RFC 3775 (Proposed Standard), June 2004.
[6] H. Fathi, S. Chakraborty, R. Prasad, Mobility management for voip: evaluation of mobile ip-based protocols, in: 2005 IEEE International Conference on Communications, ICC 2005, vol. 5, pp. 3230–3235.
[7] R. Koodli, Fast Handovers for Mobile IPv6, RFC 4068 (Experimental), July 2005.
[8] C. Castelluccia, Hmipv6: a hierarchical mobile ipv6 proposal, SIGMOBILE Mobile Computing and Communications Review 4 (2000) 48–59.
[9] A.G. Valkó, Cellular ip: a new approach to internet host mobility, SIGCOMM Computing and Communications Review 29 (1999) 50–65.
[10] R. Ramjee, K. Varadhan, L. Salgarelli, S.R. Thuel, S.-Y. Wang, T. La Porta, Hawaii: a domain-based approach for supporting mobility in wide-area wireless networks, Networking IEEE/ACM Transactions 10 (3) (2002) 396–410.
[11] Devarapalli, Proxy mobile ipv6 and mobile ipv6 interworking, Technical Report, IETF, April 2007.
[12] J.H. Saltzer, D.P. Reed, D.D. Clark, End-to-end arguments in system design, ACM Transactions on Computer Systems 2 (1984) 277–288.
[13] A.C. Snoeren, H. Balakrishnan, An end-to-end approach to host mobility, in: MobiCom'00: Proceedings of the Sixth Annual International Conference on Mobile Computing and Networking, ACM Press, New York, NY, USA, 2000, pp. 155–166.
[14] M. Riegel, M. Tuexen, Mobile sctp, <draft-riegel-tuexen-mobile-sctp-06.txt>, work in progress, March 2006.
[15] R. Stewart, Stream Control Transmission Protocol, RFC 4960 (Proposed Standard), Sept. 2007.
[16] R. Stewart, Q. Xie, M. Tuexen, S. Maruyama, M. Kozuka, Stream Control Transmission Protocol (SCTP) Dynamic Address Reconfiguration, RFC 5061 (Proposed Standard), Sept. 2007.
[17] R. Stewart, M. Ramalho, Q. Xie, M. Tuexen, P. Conrad, Stream Control Transmission Protocol (SCTP) Partial Reliability Extension, RFC 3758 (Proposed Standard), May 2004.
[18] ITU-T Recommendation G.107, The E-Model – A Computational Model in Use in Transmission Planning, March 2003.
[19] Wireless Tools for Linux. Available from: <http://www.hpl.hp.com/personal/JeanTourrilhes/Linux/Tools.html> (accessed Feb. 2008).
[20] J. Nagle, Congestion control in IP/TCP internetworks, RFC 896, Jan. 1984.
[21] V. Jacobson, Congestion avoidance and control, in SIGCOMM'88: Symposium Proceedings on Communications Architectures and Protocols, ACM, New York, NY, USA, 1988, pp. 314–329.
[22] Netem: Network Emulator. Available from: <http://www.linux-foundation.org/en/Net:Netem> (accessed Feb. 2008).
[23] H. Schulzrinne, S. Casner, R. Frederick, V. Jacobson, RTP: A Transport Protocol for Real-Time Applications, RFC 1889 (Proposed Standard), Jan. 1996 (Obsoleted by RFC 3550).
[24] ITU-T Recommendation P.862, Perceptual Evaluation of Speech Quality, 2001.
[25] M. Atiquzzaman, S. Sivagurunathan, Multimedia over wireless mobile data network mobile multimedia communications: concepts applications and challenges, Information Science Reference (2008).
[26] R. Chandra, P. Bahl, MultiNet: connecting to multiple IEEE 802.11 networks using a single wireless card, in: INFOCOM 2004, Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies, vol. 2, 7–11 March 2004, pp. 882–893.
[27] Virtual wifi: Faqs. Available from: <http://research.microsoft.com/netres/projects/virtualwifi/faq.htm>.
[28] IEEE 802.16e-2005, IEEE Standard for Local and Metropolitan Area Networks – Part 16: Air interface for Fixed Broadband Wireless Access systems – Amendment 2: Physical and Medium Access Control layers for combined fixed and mobile operation in licensed bands and Corrigendum 1, February 2006.
[29] J. Iyengar, P. Amer, R. Stewart, Concurrent multipath transfer using SCTP multihoming over independent end-to-end paths, IEEE/ACM Transactions on Networking 14 (5) (2006) 951–964.
[30] J. Iyengar, P. Amer, R. Stewart, Retransmission policies for concurrent multipath transfer using SCTP multihoming, in: Proceedings, 12th IEEE International Conference on Networks, ICON 2004, vol. 2, 16–19 Nov. 2004, pp. 713–719.
[31] Psytechnics, Estimating E-Model Id within a VoIP Network, Technical Report, Psytechnics, 2002.
[32] C. Cicconetti, L. Lenzini, E. Mingozzi, C. Eklund, Quality of service support in IEEE 80216 networks, Network IEEE 20 (2006) 50–55.